FRED LIVINGSTON, PHD

TEACHING STATEMENT

"I'm excited to bring my 20 years of experience in the autonomy industry and multiple patents and commercial products into academia."

My teaching philosophy is built around the idea of collaborative learning. Students learn best when working together on projects and sharing ideas. I also believe that technology can be an excellent tool for learning, especially for remote sensing and simulation. Using these tools, students can better understand the systems they are studying and how they work together.

The subject of autonomy is not new; it has been around for quite some time. At its most basic level, many disciplines are involved, including engineering and science fields like mathematics, kinematics and kinetics, electric circuit theory, electronics, communications, control, and computing. Robotics is the study of creating machines that can think, feel, and act like humans. The field requires an understanding of synthesizing human reasoning with robotic sensors to make intelligent decisions about how it should respond in any given situation. This includes using both computer vision algorithms; for tracking surrounding objects, or hearing; surrounding noise, as well as using artificial intelligence concepts; i.e., machine learning. By developing artificial intelligence robots learn from their own experience; heuristic learning, without the need to be programmed beforehand. New artificial intelligence algorithms will help robot's develop their planning skills; which are human learning skills that can adjust task goals when and where necessary.

The student should be familiar with various elements of robotic systems, such as kinematics, kinetics, and control. They will also learn how to apply these principles in an industrial or research setting by understanding cyber-physical system analysis, i.e., information from sensor data, sending messages via communication networks (e.g., 5G), and processing data using computing capability, which could include artificial intelligence techniques like machine learning mentioned earlier. I believe that an autonomy curriculum should contain the following subjects:

- Dynamics and Control
- Robotics
- Mechatronics
- Perception
- Machine Learning
- Cyber-Physical Systems
- Localization, Navigation, and Mapping
- Object Oriented Programming
- Embedded Systems

I sought out an opportunity to develop and hone my teaching skills by working as a part-time adjunct assistant professor at North Carolina State University. In the Spring semester of 2022, I taught Computer Control of Robotics to a mixed class of upper-level undergraduate students and graduate students. In that class, I focused on developing a typical curriculum; I used a combination of teaching methods, i.e., digital slides, demonstrations, and student-driven problem-solving. This included learning Homogeneous Transformation (the mathematical relationship between two object frames), Kinematics (geometric considerations without force include), Kinetics (geometric considerations with force included, i.e., dynamics). In their study of dynamics the student learned how to determine and apply, i.e., manage, forces being applied to the object under control, i.e., the robot manipulator. The student learned that an understanding dynamics was required before control strategies could be applied. This challenging course was demanding for the students. As such, it required me to adapt quickly to the needs of the students, since I discovered the student's came from a variety of backgrounds and they also had different learning styles.

In order to unify the students in the course I utilized and adapted robotics examples from my professional career. As a Principal Robotics Engineer I was able to provide students with real-life examples and the latest technology trends used in the autonomy industry, including emerging topics that included edge computing, modeling, and simulation. These tools are used in industry to assist, develop, and verify, the development of autonomous algorithm. For example, one can add real-life disturbances and hazards to the environment. Today, many of the developed autonomous systems are designed to improve the quality of life of human operators by making their environment safe. Students enhance their analytical or mathematical solutions for intelligent automation and manufacturing, including robotics, by using simulation tools such as MATLAB, Robotic Operation System (ROS2), and NVIDIA ISAAC platform for intelligent automation.

To ensure a successful engineering career an engineering curriculum must provide students with a firm foundation in science, technology, and mathematics. To ensure a successful career in robotics and automation, students must apply their knowledge by undertaking "hands-on" laboratories and projects with real robotic systems. Simulation has always been an essential part of this process, but simulation is only effective when combined with practical laboratories and assignments. Providing students with real-time access to cyber-physical systems locally, or remotely, is a key element to learning state-of-the-art engineering. Students must be able to make a reasoned judgment when selecting an appropriate automation and robotic systems for a given application, and they understand the limitations of such a system.

I studied Electrical and Computer Engineering as an undergraduate. As a graduate student I studied robotics, mechatronics, mechanical design, and 3D printing I have been fortunate to have had these enjoyable professional and educational experiences. These experiences have shaped my philosophy as to what constitutes the skills required by an “engineer” in today’s world. I have developed my teaching philosophy and style based on that understanding. I want to contribute more to the teaching and training of the “engineers” of today. I am particularly interested in teaching both undergraduate and graduate-level in the following courses:

- Computer Control of Robotics Systems
- Planning for Autonomous Systems
- Robot Modeling
- Perception for Autonmous Robots
- Mechatronics
- Object-Orientated Programming (C++, Python)
- Middleware and Architecture for Cyber-Physical Systems